3. (Unchanged) The system of claim 1, wherein said primary data structure comprises a pointer to a secondary data structure.

4. (Unchanged) The system of claim 1, further comprising a node identifier for each node for identifying positional placement of a resource.

5. (Unchanged) The system of claim 4, wherein said node identifier represents multiple levels of interconnect.

6. (Unchanged) The system of claim 1, further comprising a dynamic updator of at least the first and second descriptors.

7. (Unchanged) The system of claim 6, wherein said dynamic updator reflects real-time system configuration into the first descriptor.

8. (Unchanged) The system of claim 6, wherein said dynamic updator reflects real-time system performance into the second descriptor.

9. (Unchanged) The system of claim 1, wherein said second descriptor is selected from the group consisting of: processor descriptors, bus descriptors, memory descriptors and shared cache descriptors.

10. (Amended) The system of claim 9, wherein said shared cache descriptor reflects [interconnect] <u>interconnects</u> of the system.

11. (Unchanged) The system of claim 10, wherein said shared cache descriptor reflects latencies of the interconnects.

12. (Unchanged) The system of claim 1, wherein said first descriptor reflects average latency between the node groups.

13. (Amended) An article comprising:

a computer-readable signal bearing medium readable by a computer having multiple processors and a plurality of resources assigned to node groups;

means in the medium for determining topological levels of at least some of the resources; and

means in the medium for determining performance of said resources ;

wherein said topological level determining means and said performance determining means are capable of being stored in firmware of the system.

14. (Unchanged) The article of claim 13, wherein the medium is a recordable data storage medium.

15. (Unchanged) The article of claim 13, wherein the medium is a modulated carrier signal.

16. (Unchanged) The article of claim 13, wherein said topological level determining means is a first descriptor and said performance determining means is a second descriptor.

17. (Unchanged) The article of claim 13, further comprising a node identifier for identifying positional placement of a resource for each node.

18. (Unchanged) The article of claim 16, wherein said second descriptor is selected from the group consisting of: processor descriptors, bus descriptors, memory descriptors and shared cache descriptors.

19. (Unchanged) The article of claim 13, wherein said second descriptor comprises a shared cache descriptor which reflects interconnect of resources.

20. (Unchanged) The article of claim 19, wherein said shared cache descriptor reflects latencies of the interconnects.

21. (Unchanged) The system of claim 16, wherein said second descriptor reflects average latencies between node groups.

22. (Amended) A method for enabling allocation of resources in a multiprocessor, comprising:

assigning multiple resources into node groups; and

maintaining system resource topology and performance descriptions as at least one data structure produced by firmware .

23. (Unchanged) The method of claim 22, further comprising

traversing the data structure to enable allocation of at least some of the resources.

24. (Unchanged) The method of claim 22, wherein said traversal step includes accessing a second data structure.

25. (Unchanged) The method of claim 24, wherein said secondary data structure is selected from the group consisting of: processor descriptors, bus descriptors, memory descriptors and shared cache descriptors.

26. (Unchanged) The method of claim 24, wherein said secondary data structure includes a shared cache descriptor for describing at least part of a system interconnect including latency between sibling nodes.

27. (Unchanged) The method of claim 22, further comprising maintaining at least average latency between at least two of the nodes.